Shack Signs

Thanks for purchasing a Shack Sign! I hope it makes a nice glowy addition to your QTH. It can be powered with any standard USB charger. I didn’t include one because most of us have dozens of these in the junk drawer from old cell phones, etc. The default rainbow animation consumes approx. 150mA, which any charger or computer USB port can easily provide.

The electronics driving the Shack Sign consists of 12 WS2812B addressable RGB LEDs (commonly known as ‘NeoPixels’), driven by Digispark clone microcontroller. Yes, this is over-kill for making a simple glowing sign, but it does make your Shack Sign very hackable! If you have no interest in customizing your Shack Sign, no need to read further other than to note the Digispark is an actual computer that has typical computer silliness. If the animation isn’t running smoothly or freezes, simply reboot it by momentarily unplugging the USB charger.

Hacking your Shack Sign

The Digispark is an Arduino compatible Attiny85 based microcontroller. Step-by-step instructions for adding Digispark board support to your Arduino IDE can be found at: <http://digistump.com/wiki/digispark/tutorials/connecting>

The NeoPixel animations are handed by the fantastic Adafruit NeoPixel library. Step-by-step instructions for adding the library to your Arduino IDE can be found at: <https://learn.adafruit.com/adafruit-neopixel-uberguide/>

Lastly, the default code running on your Shack Sign, which is shamelessly based on Adafruit NeoPixel example code, can be found at: <https://github.com/3d0g/Shack-Sign>

**A word about power:** NeoPixels have red, green and blue LEDs inside that consume about 20mA per color. This means if you were to set all 12 NeoPixels to white, full brightness, the power requirement would be over 720mA which exceeds what most computer USB ports can provide. Now computers are usually smart enough to shut down the port when current capacity is exceeded but there’s no guarantees! Don’t kill your computer! Always set the BRIGHTNESS constant in the code to 128 or less while developing your code. Using an externally powered USB hub can also provide an additional safety measure and may be required if your project’s current requirements are high. Power for the NeoPixels come from the CON3 connector on the Digispark, which is direct-wired to the micro-USB connector. This means if your project does need more than 500mA you can use a high-capacity USB charger without issue. Just remember the limitations of your computer!

**I/O pins:** The Digispark has 6 I/O pins. Pin 0 is used to drive the NeoPixels, leaving the rest at your disposal. Pin 1 has an on-board LED attached which can be handy for code debugging. Note pins 3 & 4 are also used for USB communications so you may need some enable/disable switching for any electronics you connect here to allow code uploads. The Digispark Wiki covers the nitty gritty on I/O specifics and limitations. Realize while the Digispark is Arduino compatible, it’s not a full Arduino board and doesn’t have the hardware protections of one. It’s very easy to let out the ‘magic smoke’ with a mis-coded output vs input, reverse polarity, soldering shorts etc. I’ve even killed one with a hearty static zap.

**USB communications:** It is possible to create an I/O connection between your computer and the Digispark. Note the Digispark doesn’t have an actual USB controller on-board but mimics one via software. It’s a very clever hack but has limitations. The Digispark Wiki has the specifics on establishing USB connectivity. The potential here is great though – make your Shack Sign change colors based on band conditions. Make it flash when a rare FT8 contact appears. The possibilities are endless.

**Questions?** Give me a shout! [jeff@3d0g.net](mailto:jeff@3d0g.net)

73, N0KAI

**...-.-**

